REMARKS

Claims 1-3, 5, 7-9, 11-22, 24-27, 29-31, 35, 36, and 39-45 are currently pending in the subject application and are presently under consideration. Claims 1, 8 and 22 have been amended as shown on pages 2-6 of Reply. Claims 6 and 23 have been cancelled. In addition, claims 44 and 45 have been newly added.

Applicants' representative thanks the Examiner for the courtesies extended during the teleconference of August 26, 2008.

Favorable reconsideration of the subject patent application is respectfully requested in view of the comments and amendments herein.

I. Rejection of Claims 1-3, 5, 7, and 27 Under 35 U.S.C. §102(b)

Claims 1-3, 5, 7, and 27 stand rejected under 35 U.S.C. §102(b) as being anticipated by Sherrod (US 4,642,756). Withdrawal of this rejection is requested since Sherrod fails to teach or suggest all aspects of subject claims.

> A single prior art reference anticipates a patent claim only if it ***expressly or inherently describes each and every limitation*** set forth in the patent claim. *Trintec Industries, Inc. v. Top-U.S.A. Corp.,* 295 F.3d 1292, 63 USPQ2d 1597 (Fed. Cir. 2002); *See Verdegaal Bros. v. Union Oil Co. of California,* 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). ***The identical invention must be shown in as complete detail as is contained in the ... claim***. *Richardson v. Suzuki Motor Co.,* 868 F.2d 1226, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Applicants' claimed subject matter relates to systems and methods for specifying and executing temporal order events. To this end, independent claim 1 recites *a constraint component that receives loose temporal constraints associated with a plurality of events..., a system information component that receives execution system information...; and an order component that determines... a plurality of event orders in accordance with the loose temporal constraints* ***and selects an optimal event order based at least in part on execution system information, wherein the event order specifies the execution order of events***. Sherrod neither teaches nor suggests such novel aspects with respect to, *an order component that determines... a plurality of event orders in accordance with the loose temporal constraints and selects an* *optimal event order based at least in part on execution system information, wherein the event order specifies the execution order of events.*

Sherrod provides method and apparatus for scheduling the execution of a plurality of processing tasks within a computer system. A task scheduler schedules the execution of a plurality of tasks within a computer system. The task scheduler utilizes a combination of externally assigned priorities and internally calculated priorities to optimize the responsiveness of the computer to external interactions; and this reference fails to teach or suggest each aspect of the claimed subject matter.

At page 3 of Office Action, it is erroneously asserted that Sherrod substantially teaches *a constraint component that receives loose temporal constraints associated with a plurality of events and an order component that determines a plurality of event orders in accordance with the loose temporal constraints and selects an optimal event order based at least in part on execution system information, wherein the event order specifies the execution order of events*, with respect to independent claim 1. The reference (Sherrod) provides for two priority values associated with each task stored in RAM and ROM, an internal priority provided from within the task scheduler, and an external priority assigned by the computer operator or the task itself. External task priority values are arranged in three groups *i.e.* fixed-low-priority, fixed-high-priority and interactive-priority based on three parameters, PRI-LOW, PRIHI and PRIMAX, *which are set by the computer operator*. The internal priority for a task is determined by the 'state' of the task and the state of a task depends upon the range where the priority of the task lies among the parameters PRI-LOW, PRIHI and PRIMAX chosen by the computer operator. The state for the fixed-high-priority task is S$RT and is placed at the top of the list and is executed first. The states for the interactive priority tasks are S$TTFN, S$HICP, S$IOFN and S$CPU and executed after the S$RT. The state of the fixed-low-priority task is S$LOW and is executed in last preceding only S$WAIT task which is waiting on some event to occur before execution can proceed (*See*, Col. 4, lines 31- Col. 5, line 15 & Table 2). The other cited portion of reference (Sherrod) provides for examining state of a task at the top of an ordered list of tasks. If the state of that task is S$WAIT, no task is executed and examination of the state of the task at the top of the list is continued until it becomes other than S$WAIT. If the state of the task at the top of the list is other than S$WAIT, that task is executed until it is no longer the task at the top of the list or its state changes to S$WAIT wherein examination of the state of the task at the top of the list is continued (*See*, Col. 5, lines 60-67). Hence Sherrod provides for preparing a list of tasks sorted in descending order by internal priority, examining the state of a task at the top of the list and executing that task if the state of that task is other than S$WAIT. The state of the task and the list for internal priority of each task in descending order is prepared according to the range where the priority of the task lies among the parameters PRI-LOW, PRIHI and PRIMAX chosen by the computer operator. Hence the operator chooses parameters PRI-LOW, PRIHI and PRIMAX and decides if a task belongs to fixed-high-priority, interactive priority or fixed-low-priority group. The priority group decides the state of the task (SRT, STTFN, S$HICP, S$IOFN, SCPU, SLOW and S&WAIT) according to seven parameters (QUAN0, QUAN1, QUAN1A… QUAN3) *chosen by the computer operator* and the state of the task decides internal priority which determines the execution order of tasks. Hence Sherrod provides for calculating internal priority for each task which is dependent upon the state of task wherein the state is dependent upon the parameters chosen by the operator and executing a task according to internal priority. More particularly, Sherrod provides for determining execution order of tasks according to parameters chosen by the operator or priority decided by the operator. Once the operator has chosen parameters and assigned any priority to a task for execution order of tasks, the system provided by Sherrod, can not change execution order of tasks according to the execution system information. Further Sherrod provides for assigning a priority to a task relative to another task so that execution order of tasks is decided according to assigned priority. Nowhere does Sherrod teach receiving loose temporal constraints associated with a plurality of events, determining a plurality of different execution orders in accordance with the loose temporal constraints and selecting an optimal event order based at least in part on execution system information. Hence Sherrod fails to teach or suggest *receiving loose temporal constraints (priorities) associated with a plurality of events and determining a plurality of event orders in accordance with the loose temporal constraints and selecting an optimal event order based at least in part on execution system information, wherein the event order specifies the execution order of events*. Through this feature, the claimed subject matter facilitates selecting optimal event order for execution of a plurality of events according to temporal constraints associated with the plurality of events and the execution system information. Temporal constraints include start times, stop times, and event duration. An optimal event order is selected based on execution system information such as available memory, cache coherency, number of processors and the like (*See*, Specification, page 22, lines 23-30, Fig. 15).

At page 4 of the Office Action, it is erroneously asserted that Sherrod substantially teaches *a system information component that provides information about an execution system to the order component to facilitate selection of an optimal event order*, with respect to dependent claim 5. The reference (Sherrod) provides for arranging external task priority values in three groups *i.e.* fixed-low-priority, fixed-high-priority and interactive-priority based on three parameters, PRI-LOW, PRIHI and PRIMAX, *which are set by the computer operator* (*See*, Table 1). The internal priority for a task is determined by the 'state' of the task. The state of a task depends upon the range where the priority of the task lies among the parameters PRI-LOW, PRIHI and PRIMAX chosen by the computer operator. The state for the fixed-high-priority task is S$RT and is placed at the top of the list and is executed first. The states for the interactive priority tasks are S$TTFN, S$HICP, S$IOFN and S$CPU and executed after the S$RT. The state of the fixed-low-priority task is S$LOW and is executed in last preceding only S$WAIT task which is waiting on some event to occur before execution can proceed (*See*, Table 2). Hence Sherrod only provides for arranging external task priority in three groups according to three parameters PRI-LOW, PRIHI and PRIMAX chosen by the computer operator and arranging task in descending order of internal priority, the internal priority being dependent upon states and the state being dependent upon the group according to parameters chosen by the operator. However Sherrod does not contemplate *a system information component that provides information about an execution system to the order component to facilitate selection of an optimal event order*. Through this feature, the claimed subject matter facilitates optimizing the execution order of a series of events. Hence a user can specify a series of constraints loosely. Thus, any number of orderings for events can satisfy the constraints. Systems can choose one of the orderings as the favored ordering. For example, one system could choose to optimize the ordering of events for maximum performance. To accomplish this goal, the system utilizes heuristics or a pool of knowledge about the executing system of which a user who specified the constraints does not know. For example, the system takes into account cache coherency, data throughput, number of number of processors, and available memory.

In view of at least the foregoing, it is readily apparent that Sherrod fails to teach or suggest all aspects of the claimed subject matter. Accordingly, it is respectfully requested that this rejection of independent claims 1 and 27 (and the claims that depend there from) should be withdrawn.

II. Rejection of Claims 8, 9, 12-26, 29-31, 35, 36, and 39-42 Under 35 U.S.C. §102(b)

Claims 8, 9, 12-26, 29-31, 35, 36, and 39-42 stand rejected under 35 U.S.C. §102(b) as being anticipated by Jerome *et al.* (US 6,323,882). Withdrawal of this rejection is requested since reference (Jerome *et al.*) fails to teach or suggest all aspects of subject claims.

> A single prior art reference anticipates a patent claim only if it ***expressly or inherently describes each and every limitation*** set forth in the patent claim. *Trintec Industries, Inc. v. Top-U.S.A. Corp.,* 295 F.3d 1292, 63 USPQ2d 1597 (Fed. Cir. 2002); *See Verdegaal Bros. v. Union Oil Co. of California,* 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). ***The identical invention must be shown in as complete detail as is contained in the ... claim***. *Richardson v. Suzuki Motor Co.,* 868 F.2d 1226, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Applicants' claimed subject matter relates to systems and methods for specifying and executing temporal order events. To this end, independent claim 8 recites *a display component that provides a plurality of object workspaces,* ***the workspaces are user interfaces including at least one of a past, present and/or future space****, the present space is an editable area,* ***the past and future space specify temporal constraints associated with a plurality of events****; and a design component that temporally associates and/or disassociate objects in the editable area and determines an optimal execution order of events based at least in part on the object associations specifying temporal constraints* ***wherein non-associated objects order of execution is determined via utility-based analysis***. Jerome *et al.* neither teaches nor suggests such novel aspects.

Jerome *et al.* provides method and system for creating and using a graphical task scheduler. A Graphical User Interface (GUI) allows a user to graphically build a real time flow task scheduler by providing a "click & drag" functional palette which contains graphical flow chart elements. By creating this graphical flow chart, the user is able to associate a plant layout, or any subpart of the entire plant, to its associated mathematical model. The scheduler controls the running of each sequence based on the scheduling information. Once a plant has been optimized by the graphical task scheduler, the optimized data may be used to control plant operations.

At page 5 of Office Action, it is erroneously asserted that Jerome *et al.* substantially teaches *a display component that provides a plurality of object workspaces, the workspaces are user interfaces including a past, present and future space*, with respect to independent claim 8. The reference (Jerome *et al.*) provides for a graphical user interface (GUI) allowing a user to construct real-time sequences. The user selects a task from a list and inserts the task into the appropriate location in the sequence. The user continues to add tasks to the sequence until the sequence is complete. For scheduled sequences, the user defines the schedule of execution. For each sequence created, a sequence process flow diagram (PFD) window is opened. In the PFD window, a unique identifying name of the sequence is displayed in the identification bar. A series of pull down menus and a button bar are provided for the user to interface with the PFD window. *The PFD window includes a sequence display window which provides a graphical display of the current sequence to the user.* The user creates a sequence by using a keyboard, a mouse, or other pointing device such as a trackball or joystick to drop and drag tasks from the task palette into the sequence display window. To add the task to the current sequence, the user drags the task into the proper location in the sequence display window. The sequence display window provides a continual visual display to the user of the tasks included in the current sequence. An example of a process of creating and modifying a sequence using the sequence PFD window is illustrated here. An initial sequence is created which includes a start state followed by a Task A and a Task B. An exit branch of Task B is a terminal exit branch which causes the sequence to stop running. The user selects and drags a third Task C having two terminal exit branches and into the sequence display window. The initial sequence is connected to the new Task C by a line. By connecting Task C to Task B, the terminal exit branch of Task B is deleted (*See*, Col. 5, line 45 – Col. 6, line 37). Hence Jerome *et al.* only provides for a graphical user interface allowing a user to construct real time sequences by selecting a task from a list and inserting the task into the appropriate location in the sequence. More particularly, Jerome *et al.* provides for only a sequence display window which provides a graphical display of *the current sequence* to the user and a constant indication of when the sequence is scheduled to run next. However, Jerome *et al.* nowhere teaches or suggests *a display component that provides a plurality of object workspaces, the workspaces are user interfaces including at least one of a* *past, present and/or future space*. Through this feature, the claimed subject matter facilitates providing a context and serving as navigational aids for a user during application development. The past and future spaces contain simplified representations of events that happened before or after the current action, respectively and also facilitate specification of temporal constraints (*See*, Specification, page 10, line 11- page 11, line 10).

At page 9 of the Office Action, it is erroneously asserted that Jerome *et al.* substantially teaches that *non-associated objects order of execution is determined via utility-based analysis*, with respect to independent claim 8. The cited portion of reference (Jerome *et al.*) provides for creating a sequence by a user. The user opens a sequence process flow diagram (PFD) window. In the PFD window, a unique identifying name of the sequence is displayed in the identification bar. A series of pull down menus and a button bar are provided for the user to interface with the PFD window. *The PFD window includes a sequence display window which provides a graphical display of the current sequence to the user.* The user creates a sequence by using a keyboard, a mouse, or other pointing device such as a trackball or joystick to drop and drag tasks from the task palette into the sequence display window. *To add the task to the current sequence, the user drags the task into the proper location in the sequence display window.* The sequence display window provides a continual visual display to the user of the tasks included in the current sequence (*See*, Col. 5, lines 45-67). Hence Jerome *et al.* provides for creating a sequence by a user and adding tasks in the sequence into the proper location in the sequence display window. More particularly, Jerome *et al.* provides for scheduling the sequences and executing the sequences only according to the schedule specified the user. However Jerome *et al.* does not contemplate that *non-associated objects order of execution is determined via utility-based analysis*. Through this feature, the claimed subject matter facilitates optimizing the execution order of a series of events by performing utility based analysis. Hence a user can specify a series of constraints loosely. Thus, any number of orderings for events can satisfy the constraints. Systems can choose one of the orderings as the favored ordering. For example, one system could choose to optimize the ordering of events for maximum performance. To accomplish this goal, the system utilizes heuristics or a pool of knowledge about the executing system of which a user who specified the constraints does not know. For example, the system takes into account cache coherency, data throughput, number of number of processors, and available memory (*See*, Specification, Page 12, lines 1-19).

At page 10 of the Office Action, it is erroneously asserted that Jerome *et al.* substantially teaches *a query component that searches for events that satisfy a query, and displays objects associated with the events in temporal order*, with respect to dependent claim 20. The cited portion of reference (Jerome *et al.*) provides for user selecting tasks from one of the several task palettes. To add the task to a current sequence, the user drags the task into the proper location in the sequence display window. The task palette is divided into several categories. For example, in a general task window, basic tasks such as an input task, an output task and a custom task are displayed. In a model task window, tasks appear that would be used for a model application sequence. These are tasks such as load case, store case, and solve, which would not be used to control a plant but would be used to simulate the control of a plant. The generic task window displays tasks used during generic sequences (*See*, Col., 5, line 63 – Col. 6, line 11). Hence Jerome *et al.* only provides for various task palettes including general task window and model task window for selecting a task and entering in a sequence. However Jerome *et al.* does not contemplate *a query component that searches for events that satisfy a query, and displays objects associated with the events in temporal order*. Through this feature, the claimed subject matter facilitates a user to view particular objects of interest in an easy and powerful manner and providing context for the present work. The user can thus focus on just a subset of the objects he cares about, and the query component can then automatically generate the context for the work.

At page 14 of the Office Action, it is erroneously asserted that Jerome *et al.* substantially teaches that *the fuzzy edges on at beginning of the bar indicate an unspecified start time and the fuzzy logic on at end of the bar indicates an unspecified end time and/or duration*, with respect to dependent claim 41 and *hard bold edges on the bar specifies specific start and/or stop time*, with respect to dependent claim 42. The reference (Jerome *et al.*) provides for assisting the user in creating sequences on PFD window by providing feedback to the user on the status of the sequence. Each task is visually coded to indicate to the user the state of the task. In one example, the coding is a color code. For example, if a task is not fully specified or has no entry point, the color of the task border may be red. If a task is fully specified and ready to run, the task border would be black. A task which has been determined to be inactive may be bordered in gray. Visual coding is also useful during execution of the sequence to indicate to the user the status of the sequence execution. Upon initialization of the sequence, all the task borders are set to white. When a task in a sequence is currently running, its border is set to green. After successful execution of a task, the border is changed to blue. If a task was terminated abnormally, the border may be set to red indicating error for the user to investigate (*See*, Col. 6, lines 52-67). Hence Jerome *et al.* provides for only employing colors to indicate status of the sequence and task to a user. However Jerome *et al.* does not contemplate *the fuzzy edges on at beginning of the bar and at end of the bar indicate an unspecified start time and/or duration* and *hard bold edges on the bar specifies specific start and/or stop time*.

In view of at least the foregoing, it is readily apparent that Jerome *et al.* fails to teach or suggest all aspects of the claimed invention. Accordingly, it is respectfully requested that this rejection of independent claims 8 and 29 (and the claims that depend there from) should be withdrawn.

III. Rejection of Claims 6 and 43 Under 35 U.S.C. §103(a)

Claims 6 and 43 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Sherrod (US 4,642,756), in view of Howie *et al.* (US 5,093,794). Withdrawal of this rejection is requested for at least the following reasons. Claims 6 and 43 depend from independent claim 1. As stated *supra*, Sherrod do not disclose or suggest every limitation set forth in the subject independent claims. Howie *et al.* fails to make-up for the aforementioned deficiencies of the base combination. Withdrawal of this rejection is therefore respectfully requested.

IV. Rejection of Claim 11 Under 35 U.S.C. §103(a)

Claim 11 stands rejected under 35 U.S.C. §103(a) as being unpatentable over Jerome *et al.* (US 6,323,882), in view of Green *et al.* (US 4,646,231). Withdrawal of this rejection is requested for at least the following reasons. Claim 11 depend from independent claim 8. As stated *supra*, Jerome *et al.* do not disclose or suggest every limitation set forth in the subject independent claims. Green *et al.* fails to make-up for the aforementioned deficiencies of the base combination. Withdrawal of this rejection is therefore respectfully requested.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments and amendments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP543US].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,
AMIN, TUROCY & CALVIN, LLP

/Marisa Joy Zink/
Marisa Joy Zink
Reg. No. 48,064

AMIN, TUROCY & CALVIN, LLP
24TH Floor, National City Center
1900 E. 9TH Street
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731